

SEARCH REQUEST FORM

Scientific and Technical Information Center

90

Requester's Full Name: Kenneth A. Gross Examiner #: 79752 Date: 1/27/04
 Art Unit: 2122 Phone Number 305-0542 Serial Number: 09/713,633
 Mail Box and Bldg/Room Location: CPK2 5407 Results Format Preferred (circle): PAPER DISK E-MAIL

If more than one search is submitted, please prioritize searches in order of need.

Please provide a detailed statement of the search topic, and describe as specifically as possible the subject matter to be searched. Include the elected species or structures, keywords, synonyms, acronyms, and registry numbers, and combine with the concept or utility of the invention. Define any terms that may have a special meaning. Give examples or relevant citations, authors, etc, if known. Please attach a copy of the cover sheet, pertinent claims, and abstract.

Title of Invention: Minimum Delta Generator for program binaries

Inventors (please provide full names): Ramarathnam Venkatesan; Saurabh Sinha

Earliest Priority Filing Date: 11/14/2000

For Sequence Searches Only Please include all pertinent information (parent, child, divisional, or issued patent numbers) along with the appropriate serial number.

The application relates to patching a program by making a match between certain parts of the program with another program to make both programs equal to one another. This matching is done by representing programs as CFG's or Control flow graphs. Nodes or blocks of the CFG are matched and unmatched blocks are removed or merged into another representation. Neighborhoods are computed in CFG graphs and nodes are removed by neighborhood. Neighborhood is a certain set of blocks surrounding a given block. Labels are given to the blocks to help comparison. This entire process is called generating a delta. Key terms are: patching, delta, CFG, neighborhood, outlier, label, breadth first traversal, matching.

Try a thesis focus search.

Thanks!

STAFF USE ONLY

Searcher: David Holloway
 Searcher Phone #: 305-7794
 Searcher Location: CPK2 4B30
 Date Searcher Picked Up: 1-27-04
 Date Completed: 1-28-04
 Searcher Prep & Review Time: 70
 Clerical Prep Time:
 Online Time: 2 35

Type of Search

Vendors and cost where applicable

NA Sequence (#)	STN
AA Sequence (#)	Dialog <u>\$1107 7/10</u>
Structure (#)	Questel/Orbit
Bibliographic	Dr. Link
Litigation	Lexis/Nexis
Fulltext	Sequence Systems
Patent Family	WWW/Internet
Other	Other (specify)

Set	Items	Description
S1	1910775	PATCH? OR UPDAT? OR UPGRAD? OR UP() (DAT? OR GRAD?) OR REV- S? OR REPAIR? OR FIX OR FIXING OR FIXED
S2	1467018	APPLICATION? OR PROGRAM? OR SOFTWARE? OR APP OR MODULE? OR BINARY OR EXECUTABLE() CODE?
S3	3049393	DELTA OR DELTAS OR CHANG? OR DIFFER? OR VARIATION? OR VARI- ANC? OR VARY?
S4	892321	CFG OR GRAPH? OR FLOWGRAPH? OR DIAGRAM? OR CHART?
S5	18398	(NEIGHBOR? OR NEIGHBOUR? OR ADJACENT? OR NEAR? OR CONTIGUO- US? OR SURROUND?) (5N) (BLOCK? ? OR NODE? OR MODULE?)
S6	1410605	IDENTIF? OR COMPAR? OR CONTRAST? OR MATCH? OR UNMATCH?
S7	1864356	IDENTICAL? OR SIMILAR? OR SAME? OR EQUIVALENT?
S8	1261760	SOURCE? OR TARGET?
S9	3250	S1 AND S2 AND S3 AND S4
S10	2	S9 AND S5 AND (S6 OR S7)
S11	530	S1(3N)S2 AND S3 AND S4
S12	142	S11 AND (S6 OR S7)
S13	41	S11 AND S8
S14	43	S10 OR S13
S15	26	S14 AND IC=G06F?
S16	26	IDPAT (sorted in duplicate/non-duplicate order)
S17	25	IDPAT (primary/non-duplicate records only)
S18	242	S3 AND S4 AND S5
S19	72	S18 AND S2
S20	23	S18 AND S1
S21	28	(S19 OR S20) AND (S6 OR S7)
S22	8	(S19 OR S20) AND S9
S23	32	(S21 OR S22) NOT S14
S24	0	S23 AND IC=G06F-009?
S25	9	S23 AND IC=G06F?
S26	9	IDPAT (sorted in duplicate/non-duplicate order)
S27	9	IDPAT (primary/non-duplicate records only)

File 347:JAPIO Oct 1976-2003/Sep(Updated 040105)

(c) 2004 JPO & JAPIO

File 350:Derwent WPIX 1963-2004/UD,UM &UP=200406

(c) 2004 Thomson Derwent

27/5/1 (Item 1 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

015837716 **Image available**
WPI Acc No: 2003-899920/200382
XRPX Acc No: N03-718344

Mail piece data parsing and identifying method in electronic statement presentment, involves comparing match codes in contiguous blocks of print stream data to identify whether parsed mail piece data belong to same or different sets

Patent Assignee: PITNEY BOWES INC (PITB)

Inventor: SHEA M

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 20030196175	A1	20031016	US 2002124792	A	20020416	200382 B

Priority Applications (No Type Date): US 2002124792 A 20020416

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
US 20030196175	A1	8	G06F-017/00	

Abstract (Basic): US 20030196175 A1

NOVELTY - The blocks of print stream data corresponding to document pages are **identified** after reading the print stream. The mail piece data are parsed from blocks of print stream data and the **match** codes are **identified** within bar code data. The **match** codes are **compared** in the **contiguous blocks** of print stream data. The parsed mail piece data are **identified** to belong to the **same** or **different** sets of mail piece data based on the result of **matching** .

USE - For parsing and extracting data from electronic print stream in electronic bill presentment and payment (EBPP) and electronic statement presentment (ESP) **applications** .

ADVANTAGE - Integrity and classification of collected data are enhanced by consulting mail piece assembly data and page information included in legacy print stream bar code information.

DESCRIPTION OF DRAWING(S) - The figure shows the block **diagram** of structure of print stream delivery architecture.

printer (104)
physical inserter (106)
electronic inserter (110)
message router (112)
web server (116)
pp; 8 DwgNo 1/2

Title Terms: MAIL; PIECE; DATA; PARSE; **IDENTIFY** ; METHOD; ELECTRONIC; STATEMENT; **COMPARE** ; **MATCH** ; CODE; CONTIGUOUS; BLOCK; PRINT; STREAM; DATA; **IDENTIFY** ; MAIL; PIECE; DATA; BELONG; SET

Derwent Class: T01; T04; T05

International Patent Class (Main): G06F-017/00

File Segment: EPI

Set Items Description
S1 1524455 PATCH? OR UPDAT? OR UPGRAD? OR UP() (DAT? OR GRAD?) OR REVI-
S? OR REPAIR? OR FIX OR FIXING OR FIXED
S2 8031184 APPLICATION? OR PROGRAM? OR SOFTWARE? OR APP OR MODULE? OR
BINARY OR EXECUTABLE() CODE?
S3 12552680 DELTA OR DELTAS OR CHANG? OR DIFFER? OR VARIATION? OR VARI-
ANC? OR VARY?
S4 1596652 CFG OR GRAPH? OR FLOWGRAPH? OR DIAGRAM? OR CHART?
S5 13821 (NEIGHBOR? OR NEIGHBOUR? OR ADJACENT? OR NEAR? OR CONTIGUO-
US? OR SURROUND?) (5N) (BLOCK? ? OR NODE? OR MODULE?)
S6 9328238 IDENTIF? OR COMPAR? OR JUDGE? OR JUDGING OR CONTRAST? OR M-
ATCH? OR UNMATCH?
S7 4631714 IDENTICAL? OR SIMILAR? OR SAME? OR EQUIVALENT?
S8 2531384 SOURCE? OR TARGET?
S9 0 S1(3N)S2 AND S3 AND S4 AND S5
S10 10 S1 AND S2 AND S3 AND S4 AND S5
S11 407 (S6 OR S7) (3N) (S3 OR S4) AND S1(3N)S2
S12 21 (S6 OR S7) (3N) (S3 OR S4) AND S1 AND (S9 OR S5)
S13 49 S3 AND S4 AND S11
S14 76 S16 OR S10 OR S12 OR S13
S15 57 RD (unique items)
S16 52 S15 NOT PY>2001
S17 52 S16 NOT PD>20011114
S18 52 S17 NOT CY>2001
File 8:Ei Compendex(R) 1970-2004/Jan W3
(c) 2004 Elsevier Eng. Info. Inc.
File 35:Dissertation Abs Online 1861-2004/Dec
(c) 2004 ProQuest Info&Learning
File 65:Inside Conferences 1993-2004/Jan W4
(c) 2004 BLDSC all rts. reserv.
File 2:INSPEC 1969-2004/Jan W3
(c) 2004 Institution of Electrical Engineers
File 94:JICST-EPlus 1985-2004/Jan W3
(c)2004 Japan Science and Tech Corp(JST)
File 111:TGG Natl.Newspaper Index(SM) 1979-2004/Jan 26
(c) 2004 The Gale Group
File 233:Internet & Personal Comp. Abs. 1981-2003/Sep
(c) 2003 EBSCO Pub.
File 144:Pascal 1973-2004/Jan W3
(c) 2004 INIST/CNRS
File 34:SciSearch(R) Cited Ref Sci 1990-2004/Jan W3
(c) 2004 Inst for Sci Info
File 99:Wilson Appl. Sci & Tech Abs 1983-2004/Dec
(c) 2004 The HW Wilson Co.
File 95:TEME-Technology & Management 1989-2004/Jan W2
(c) 2004 FIZ TECHNIK

18/5/11 (Item 2 from file: 35)
DIALOG(R)File 35:Dissertation Abs Online
(c) 2004 ProQuest Info&Learning. All rts. reserv.

01157575 ORDER NO: AAD91-13963

A GRAPH THEORETIC APPROACH TO SCENE MATCHING (ASSOCIATION GRAPH)

Author: CHIPMAN, LAURE J.

Degree: PH.D.

Year: 1990

Corporate Source/Institution: THE UNIVERSITY OF ALABAMA IN HUNTSVILLE (0278)

Source: VOLUME 51/12-B OF DISSERTATION ABSTRACTS INTERNATIONAL.

PAGE 5969. 156 PAGES

Descriptors: COMPUTER SCIENCE

Descriptor Codes: 0984

The ability to match two scenes is a fundamental requirement in a variety of computer vision tasks. This dissertation presents a graph theoretic approach to inexact scene matching which is useful in dealing with problems due to imperfect image segmentation. A scene is described by a set of graphs, with nodes representing objects and arcs representing relationships between objects. Each node has a set of values representing various attribute measurements of the object it represents. Each arc has values representing the relations between pairs of objects, such as angle, adjacency, or distance. With this method of scene representation, the task in scene matching is to **match** two sets of **graphs**. Because of segmentation errors, variations in camera angle, illumination, and other conditions, an exact match between the sets of observed and stored graphs is usually not possible.

In the approach developed, first the problem is represented as an association graph, in which each node represents a possible mapping of an observed region to a stored object, and each arc represents the compatibility of two mappings. Nodes and arcs have weights indicating the merit of a region-object mapping and the degree of compatibility between two mappings. A **match** between the two **graphs** corresponds to a clique, or fully connected subgraph, in the association graph. The task is to find the clique that represents the best match. Fuzzy relaxation is used to **update** the node weights using the contextual information contained in the arcs and **neighboring nodes**. This simplifies the evaluation of cliques. A method of handling oversegmentation and undersegmentation problems is also presented. The approach is tested with a set of realistic images which exhibit many types of segmentation errors.

18/5/21 (Item 5 from file: 2)
DIALOG(R)File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

5519219 INSPEC Abstract Number: C9704-6110B-025

Title: A mechanism for automatically and dynamically changing software components

Author(s): Maruyama, K.; Shima, K.-I.

Author Affiliation: NTT Software Labs., Japan

Journal: Transactions of the Information Processing Society of Japan
vol.37, no.12 p.2334-51

Publisher: Inf. Process. Soc. Japan,

Publication Date: Dec. 1996 Country of Publication: Japan

CODEN: JSGRD5 ISSN: 0387-5806

SICI: 0387-5806(199612)37:12L.2334:MADC;1-0

Material Identity Number: T205-97002

Language: Japanese Document Type: Journal Paper (JP)

Treatment: Practical (P); Theoretical (T)

Abstract: In source code reuse, reusable software components must be frequently modified to create the required **programs**, since they take **fixed** codes in a library. Not all components to be reused can be prepared in a library before reusing, since the software development domain is difficult to specify. We have developed a new mechanism for automatically and dynamically **changing** software components that are called active components. These active components have the capability of modifying themselves into codes that meet users' requirements and that are in accordance with their existing environments. Therefore, the active components do not require many user modifications when they are composed, and can be reused in unspecified or floating domains. The mechanism of active components provides two kinds of **changes** by: decomposing their functions; and partially exchanging their functions with modification histories of other active components. The mechanism is achieved by a program integration algorithm based on program slicing and labeled **graph matching**. Our proposed algorithm can be applied to more kinds of source codes by introducing the complements of slices, label abstraction, and reconnection of dependence edges than the conventional algorithms can. This paper describes the new mechanism and the algorithm that achieves the mechanism, and gives examples of **changing** active components. (19 Refs)

Subfile: C

Descriptors: **graph** theory; software libraries; software reusability

Identifiers: software component **changing**; source code reuse; reusable software components; software library; software development; active components; user requirements; user modifications; program integration algorithm; program slicing; labeled **graph matching**; label abstraction

Class Codes: C6110B (Software engineering techniques); C1160 (Combinatorial mathematics)

Copyright 1997, IEE

18/5/21 (Item 5 from file: 2)

DIALOG(R)File 2:INSPEC

(c) 2004 Institution of Electrical Engineers. All rts. reserv.

5519219 INSPEC Abstract Number: C9704-6110B-025

Title: **A mechanism for automatically and dynamically changing software components**

Author(s): Maruyama, K.; Shima, K.-I.

Author Affiliation: NTT Software Labs., Japan

Journal: Transactions of the Information Processing Society of Japan
vol.37, no.12 p.2334-51

Publisher: Inf. Process. Soc. Japan,

Publication Date: Dec. 1996 Country of Publication: Japan

CODEN: JSGRD5 ISSN: 0387-5806

SICI: 0387-5806(199612)37:12L.2334:MADC;1-O

Material Identity Number: T205-97002

Language: Japanese Document Type: Journal Paper (JP)

Treatment: Practical (P); Theoretical (T)

Abstract: In source code reuse, reusable software components must be frequently modified to create the required **programs**, since they take **fixed** codes in a library. Not all components to be reused can be prepared in a library before reusing, since the software development domain is difficult to specify. We have developed a new mechanism for automatically and dynamically **changing** software components that are called active components. These active components have the capability of modifying themselves into codes that meet users' requirements and that are in accordance with their existing environments. Therefore, the active components do not require many user modifications when they are composed, and can be reused in unspecified or floating domains. The mechanism of active components provides two kinds of **changes** by: decomposing their functions; and partially exchanging their functions with modification histories of other active components. The mechanism is achieved by a program integration algorithm based on program slicing and labeled **graph matching**. Our proposed algorithm can be applied to more kinds of source codes by introducing the complements of slices, label abstraction, and reconnection of dependence edges than the conventional algorithms can. This paper describes the new mechanism and the algorithm that achieves the mechanism, and gives examples of **changing** active components. (19 Refs)

Subfile: C

Descriptors: **graph theory**; software libraries; software reusability

Identifiers: software component **changing**; source code reuse; reusable software components; software library; software development; active components; user requirements; user modifications; program integration algorithm; program slicing; labeled **graph matching**; label abstraction

Class Codes: C6110B (Software engineering techniques); C1160 (Combinatorial mathematics)

Copyright 1997, IEE

Ken Gross

5 Y07

Set	Items	Description
S1	1910775	PATCH? OR UPDAT? OR UPGRAD? OR UP() (DAT? OR GRAD?) OR REVIS? OR REPAIR? OR FIX OR FIXING OR FIXED
S2	1467018	APPLICATION? OR PROGRAM? OR SOFTWARE? OR APP OR MODULE? OR BINARY OR EXECUTABLE() CODE?
S3	3049393	DELTA OR DELTAS OR CHANG? OR DIFFER? OR VARIATION? OR VARIANC? OR VARY?
S4	892321	CFG OR GRAPH? OR FLOWGRAPH? OR DIAGRAM? OR CHART?
S5	18398	(NEIGHBOR? OR NEIGHBOUR? OR ADJACENT? OR NEAR? OR CONTIGUOUS? OR SURROUND?) (5N) (BLOCK? ? OR NODE? OR MODULE?)
S6	1410605	IDENTIF? OR COMPAR? OR CONTRAST? OR MATCH? OR UNMATCH?
S7	1864356	IDENTICAL? OR SIMILAR? OR SAME? OR EQUIVALENT?
S8	1261760	SOURCE? OR TARGET?
S9	3250	S1 AND S2 AND S3 AND S4
S10	2	S9 AND S5 AND (S6 OR S7)
S11	530	S1(3N)S2 AND S3 AND S4
S12	142	S11 AND (S6 OR S7)
S13	41	S11 AND S8
S14	43	S10 OR S13
S15	26	S14 AND IC=G06F?
S16	26	IDPAT (sorted in duplicate/non-duplicate order)
S17	25	IDPAT (primary/non-duplicate records only)
S18	242	S3 AND S4 AND S5
S19	72	S18 AND S2
S20	23	S18 AND S1
S21	28	(S19 OR S20) AND (S6 OR S7)
S22	8	(S19 OR S20) AND S9
S23	32	(S21 OR S22) NOT S14
S24	0	S23 AND IC=G06F-009?
S25	9	S23 AND IC=G06F?
S26	9	IDPAT (sorted in duplicate/non-duplicate order)
S27	9	IDPAT (primary/non-duplicate records only)
S28	128148	(S6 OR S7) (3N) (S3 OR S4)
S29	1263	S28 AND S1 AND S2
S30	7	S29 AND S5
S31	272	S6 AND S7 AND S29
S32	167	S3 AND S4 AND S29
S33	32	(S31 OR S32) AND IC=G06F-009?
S34	31	S33 NOT (S26 OR S23 OR S22 OR S14)
S35	15	S34 AND IC=(G06F-009/45 OR G06F-009/44)
S36	1	S30 AND IC=G06F-009?
S37	16	S35 OR S36
S38	16	IDPAT (sorted in duplicate/non-duplicate order)
S39	16	IDPAT (primary/non-duplicate records only)

File 347:JAPIO Oct 1976-2003/Sep(Updated 040105)

(c) 2004 JPO & JAPIO

File 350:Derwent WPIX 1963-2004/UD,UM &UP=200406

(c) 2004 Thomson Derwent

39/5/1 (Item 1 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

015594785 **Image available**

WPI Acc No: 2003-656940/200362

XRPX Acc No: N03-523363

Software patch **creating method**, involves comparing **reduced program dependency graphs** by locating functions of **graphs** and determining whether functions of **former graph** are on working stack

Patent Assignee: NORTEL NETWORKS LTD (NELE)

Inventor: MILLS P H; SCHWEITZ E A; WERLINGER D J

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 6594822	B1	20030715	US 99252738	A	19990219	200362 B
			US 2000590431	A	20000608	

Priority Applications (No Type Date): US 2000590431 A 20000608; US 99252738

A 19990219

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 6594822	B1	16		G06F-009/45	CIP of application US 99252738

Abstract (Basic): US 6594822 B1

NOVELTY - The method involves decomposing two object files into respective constituent cantles and developing a reduced **program dependency graph** for each file. The **graphs** are **compared** by locating functions of **graphs** and determining whether the functions of the former **graph** are on a working stack. The **changes** between the **graphs** are determined and a path is created based on the **changes**.

DETAILED DESCRIPTION - The functions of the latter **graphs** are **compared** with the former if the functions of the former **graphs** are not on the working stack.

An INDEPENDENT CLAIM is also included for a computer system for creating **software patches**.

USE - Used for creating **software patches**.

ADVANTAGE - The method efficiently creates a **patch** to **update** the existing **software** on a computer system without pulling the computer off-line.

DESCRIPTION OF DRAWING(S) - The drawing shows a reduced **program dependency graphs** developed from the cantles.

pp; 16 DwgNo 2/5

Title Terms: **SOFTWARE** ; **PATCH** ; **METHOD**; **COMPARE**; **REDUCE**; **PROGRAM** ; **DEPEND**; **GRAPH** ; **LOCATE**; **FUNCTION**; **GRAPH** ; **DETERMINE**; **FUNCTION**; **FORMER**; **GRAPH** ; **WORK**; **STACK**

Derwent Class: T01

International Patent Class (Main): **G06F-009/45**

File Segment: EPI

39/5/2 (Item 2 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

015494964 **Image available**
WPI Acc No: 2003-557111/200352
XRPX Acc No: N03-442745

Software tool for merging metal model versions of object oriented software system, executes difference and merge operations through interaction with data structure in each model version

Patent Assignee: RAMAKRISHNA J (RAMA-I); REDDY S S (REDD-I)

Inventor: RAMAKRISHNA J; REDDY S S

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 20030084424	A1	20030501	US 200259696	A	20020128	200352 B

Priority Applications (No Type Date): IN 2001MU722 A 20010726

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 20030084424	A1	13		G06F-009/44	

Abstract (Basic): US 20030084424 A1

NOVELTY - The tool has an interactive user display window, an object association graph used as modeled template for forming nodes and a data structure for displaying element hierarchy in compared model version (405). The user monitors the data structure in each model version from display window and executes the difference and merge operations by interaction with the data structure.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

(1) model version difference computation and reconciling method; and

(2) modeling structure.

USE - For comparing and merging versions of metal models such as unified modeling language (UML) model, meta object facility (MOF) model and extended markup language (XML) model of object oriented software system.

ADVANTAGE - Provides a software mechanism for model version comparison that provides automated difference and merge capability with dynamic update capability.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram illustrating data structure of a difference computation process.

model version (405)
pp; 13 DwgNo 4/4

Title Terms: SOFTWARE ; TOOL; MERGE; METAL; MODEL; VERSION; OBJECT; ORIENT ; SOFTWARE ; SYSTEM; EXECUTE; DIFFER ; MERGE; OPERATE; THROUGH; INTERACT; DATA; STRUCTURE; MODEL; VERSION

Derwent Class: T01

International Patent Class (Main): G06F-009/44

File Segment: EPI

39/5/10 (Item 10 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

013514558 **Image available**

WPI Acc No: 2000-686504/200067

XRPX Acc No: N00-507555

Compiled programming code translation method involves updating control flow graph based on examination of destination, and translating programming code accordingly

Patent Assignee: UNISYS CORP (BURS)

Inventor: FENDER T N; JENNINGS A T; KRABLIN G L; STRATTON W

Number of Countries: 021 Number of Patents: 006

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
WO 200045255	A2	20000803	WO 2000US2098	A	20000127	200067	B
EP 1145105	A2	20011017	EP 2000913276	A	20000127	200169	
			WO 2000US2098	A	20000127		
EP 1145105	B1	20020911	EP 2000913276	A	20000127	200264	
			WO 2000US2098	A	20000127		
JP 2002536711	W	20021029	JP 2000596446	A	20000127	200274	
			WO 2000US2098	A	20000127		
DE 60000433	E	20021017	DE 600433	A	20000127	200276	
			EP 2000913276	A	20000127		
			WO 2000US2098	A	20000127		
US 6662354	B1	20031209	US 99239282	A	19990129	200381	

Priority Applications (No Type Date): US 99239282 A 19990129

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
-----------	------	-----	----	----------	--------------

WO 200045255 A2 E 55 G06F-009/00

Designated States (National): JP

Designated States (Regional): AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE

EP 1145105 A2 E G06F-009/00 Based on patent WO 200045255

Designated States (Regional): AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE

EP 1145105 B1 E G06F-009/00 Based on patent WO 200045255

Designated States (Regional): CH DE FR GB LI

JP 2002536711 W 66 G06F-009/45 Based on patent WO 200045255

DE 60000433 E G06F-009/00 Based on patent EP 1145105

Based on patent WO 200045255

US 6662354 B1 G06F-009/45

Abstract (Basic): WO 200045255 A2

NOVELTY - A control flow graph (CFG) is constructed based on identification of basic blocks and their links. The basic blocks leading to dynamic branch, are explored based on CFG, and a set of destination addresses are determined. The destinations are examined to identify the branch data. The CFG is updated accordingly based on which the programming code is translated.

DETAILED DESCRIPTION - Several basic blocks in a code state (12) and the links between them, are identified. Based on the identification, the CFG is constructed in preliminary form. The set of destination addresses define the set of destinations from the dynamic branch. The CFG is updated to reflect the set of destinations and identified branch table, based on which the code is translated from the code state (12) to a code state (14). INDEPENDENT CLAIMS are also included for the following:

(a) translator;

(b) program product

USE - For translating compiled programming code.

ADVANTAGE - A translator which evaluates the calculation prefacing the branch, is provided in order to adjust for relocated potential destinations or different addressing structure in target instruction set.

DESCRIPTION OF DRAWING(S) - The figure shows the block diagram illustrating the translation of program from one state to another

state.

Code states (12,14)

pp; 55 DwgNo 1/16

Title Terms: **COMPILE**; **PROGRAM** ; CODE; TRANSLATION; METHOD; **UPDATE** ;
CONTROL; **FLOW**; **GRAPH** ; BASED; EXAMINATION; DESTINATION; TRANSLATION;
PROGRAM ; CODE; ACCORD

Derwent Class: T01

International Patent Class (Main): **G06F-009/00** ; **G06F-009/45**

International Patent Class (Additional): **G06F-009/32**

File Segment: EPI

39/5/14 (Item 14 from file: 350)
DIALOG(R) File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

012326395 **Image available**
WPI Acc No: 1999-132502/199911
XRPX Acc No: N99-096474

File difference extraction and patch file creation method for file updating - involves comparing old and new files to generate patches reflecting file differences and forwarding patches to be applied to other copies of the old file

Patent Assignee: POCKET SOFT INC (POCK-N)

Inventor: JONES K N

Number of Countries: 080 Number of Patents: 005

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
WO 9904336	A1	19990128	WO 98US14433	A	19980715	199911 B
AU 9883972	A	19990210	AU 9883972	A	19980715	199925
EP 1005674	A1	20000607	EP 98934459	A	19980715	200032
			WO 98US14433	A	19980715	
US 6526574	B1	20030225	US 9752584	P	19970715	200323
			WO 98US14433	A	19980715	
			US 2000463035	A	20000117	
CA 2295479	C	20031230	CA 2295479	A	19980715	200404
			WO 98US14433	A	19980715	

Priority Applications (No Type Date): US 9752584 P 19970715; US 2000463035
A 20000117

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
-----------	------	-----	----	----------	--------------

WO 9904336	A1	E	47	G06F-009/445	
------------	----	---	----	--------------	--

Designated States (National): AL AM AT AU AZ BB BG BR BY CA CH CN CU CZ
DE DK EE ES FI GB GE GH GM HR HU ID IL IS JP KE KG KP KR KZ LC LK LR LS
LT LU LV MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR
TT UA UG US UZ VN YU ZW

Designated States (Regional): AT BE CH CY DE DK ES FI FR GB GH GM GR IE
IT KE LS LU MC MW NL OA PT SD SE SZ UG ZW

AU 9883972	A	G06F-009/445	Based on patent WO 9904336
------------	---	--------------	----------------------------

EP 1005674	A1	E	G06F-009/445	Based on patent WO 9904336
------------	----	---	--------------	----------------------------

Designated States (Regional): DE GB

US 6526574	B1	G06F-009/44	Provisional application US 9752584
			Based on patent WO 9904336

CA 2295479	C	E	G06F-009/445	Based on patent WO 9904336
------------	---	---	--------------	----------------------------

Abstract (Basic): WO 9904336 A

NOVELTY - A file, e.g. application code, is available as an original version and a new version, e.g. new revision. Both file versions are input to a patch build program (23). This compares the files and finds similarities and differences. The differences are formed into a patch file (36). This file is sent to another location where the old file version still exists. A patch applying program (31) uses the patch file to update the old file to the new file.

USE - Updating application or data files to new versions

ADVANTAGE - By extracting the differences, the updating

information is reduced in size and can be transferred more efficiently.

DESCRIPTION OF DRAWING(S) - DESCRIPTION OF DRAWING(S) - Patch use system (20,22) Old and new files for comparison ; ((23) Patch file generator; (31) Patch applying program ; (40) Resulting new file.

Dwg.1a,1b/

7

Title Terms: FILE; DIFFER; EXTRACT; PATCH ; FILE; CREATION; METHOD; FILE; UPDATE ; COMPARE ; NEW; FILE; GENERATE; PATCH ; REFLECT; FILE; DIFFER; FORWARDING; PATCH ; APPLY; COPY; FILE

Derwent Class: T01

International Patent Class (Main): G06F-009/44 ; G06F-009/445

International Patent Class (Additional): G06F-009/45

39/5/16 (Item 16 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

009366967 **Image available**

WPI Acc No: 1993-060446/199308

XRPX Acc No: N93-046151

Controlling change in multiple development environments - using working tables and files to merge delta structures whenever versions are reconciled between environments

Patent Assignee: SUN MICROSYSTEMS INC (SUNM)

Inventor: SKINNER G

Number of Countries: 006 Number of Patents: 006

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
EP 528617	A2	19930224	EP 92307304	A	19920810	199308	B
EP 528617	A3	19930915	EP 92307304	A	19920810	199509	
US 5481722	A	19960102	US 91746957	A	19910819	199607	
			US 9384078	A	19930628		
			US 94338883	A	19941114		
EP 528617	B1	19991222	EP 92307304	A	19920810	200004	
DE 69230452	E	20000127	DE 630452	A	19920810	200012	
			EP 92307304	A	19920810		
KR 169327	B1	19990115	KR 9214904	A	19920819	200038	

Priority Applications (No Type Date): US 91746957 A 19910819; US 9384078 A 19930628; US 94338883 A 19941114

Cited Patents: No-SR.Pub; 4.Jnl.Ref

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
EP 528617	A2	E	31 G06F-009/44	Designated States (Regional): DE FR GB IT
EP 528617	A3		G06F-009/44	
US 5481722	A	25	G06F-015/16	Cont of application US 91746957 Cont of application US 9384078
EP 528617	B1	E	G06F-009/44	Designated States (Regional): DE FR GB IT
DE 69230452	E		G06F-009/44	Based on patent EP 528617
KR 169327	B1		G06F-009/00	

Abstract (Basic): EP 528617 A

The source **module** and its changes in each of at least two interrelated development environments are maintained in delta structures. Various procedures and working tables/files are provided to merge the delta structures whenever the latest **revision** /edition of the source **module** in one of the environments is reconciled/resynchronised to the latest edition/ **revision** of the source **module** in the other environment. As a result, all change deltas are maintained and propagated among the environments.

USE/ADVANTAGE - For example **software** development system for at least two **software** development environments without any loss of change history.

Dwg.5/8

Title Terms: CONTROL; CHANGE; MULTIPLE; DEVELOP; ENVIRONMENT; WORK; TABLE; FILE; MERGE; DELTA; STRUCTURE; VERSION; ENVIRONMENT

Derwent Class: T01

International Patent Class (Main): G06F-009/00 ; G06F-009/44 ; G06F-015/16

International Patent Class (Additional): G06F-007/22; G06F-013/00

File Segment: EPI

Set Items Description
S1 619472 PATCH? OR UPDAT? OR UPGRAD? OR UP() (DAT? OR GRAD?) OR REVI-
S? OR REPAIR? OR FIX OR FIXING OR FIXED
S2 2221687 APPLICATION? OR PROGRAM? OR SOFTWARE? OR APP OR MODULE? OR
BINARY OR EXECUTABLE() CODE?
S3 1666999 DELTA OR DELTAS OR CHANG? OR DIFFER? OR VARIATION? OR VARI-
ANC? OR VARY?
S4 593230 CFG OR GRAPH? OR FLOWGRAPH? OR DIAGRAM? OR CHART?
S5 26473 (NEIGHBOR? OR NEIGHBOUR? OR ADJACENT? OR NEAR? OR CONTIGUO-
US? OR SURROUND?) (5N) (BLOCK? ? OR NODE? OR MODULE?)
S6 949032 IDENTIF? OR COMPAR? OR JUDGE? OR JUDGING OR CONTRAST? OR M-
ATCH? OR UNMATCH?
S7 1248824 IDENTICAL? OR SIMILAR? OR SAME? OR EQUIVALENT?
S8 610336 SOURCE? OR TARGET?
S9 3411 (S6 OR S7) (3N) (S3 OR S4) (S) S1 (S) S2
S10 973 S9 (S) S8
S11 3003 (S6 OR S7) (3N) (S3 OR S4) (10N) S1 (10N) S2
S12 33 S10 (S) S5
S13 985 (S6 OR S7) (3N) (S3 OR S4) (5N) S1 (5N) S2
S14 76 S13 AND IC=(G06F-009/45 OR G06F-009/44)
S15 103197 S3 (15N) S4
S16 43 S14 AND S15
S17 34 S16 NOT AD>20001114
S18 1 S12 AND IC=G06F-009?
S19 31 S1 (2N) S2 AND S17
S20 31 S18 OR S19
S21 31 IDPAT (sorted in duplicate/non-duplicate order)
S22 30 IDPAT (primary/non-duplicate records only)
File 348:EUROPEAN PATENTS 1978-2004/Jan W04
(c) 2004 European Patent Office
File 349:PCT FULLTEXT 1979-2002/UB=20040122,UT=20040115
(c) 2004 WIPO/Univentio

00263325

Display method in software development support system.
Anzeigeverfahren in einem Unterstützungssystem zur Software-Entwicklung.
Methode de présentation d'informations dans un système d'aide au
développement de logiciels.

PATENT ASSIGNEE:

HITACHI, LTD., (204141), 6, Kanda Surugadai 4-chome, Chiyoda-ku, Tokyo
101, (JP), (applicant designated states: DE)
Hitachi Computer Engineering Co., Ltd., (772550), 1, Horiyamashita,
Hadano-shi Kanagawa-ken, (JP), (applicant designated states: DE)
Hitachi Seibu Soft Ware Company, Ltd., (790470), 4-6, Kitahama Higashi-ku
, Osaka, (JP), (applicant designated states: DE)

INVENTOR:

Naito, Ichiro, 12-10-503, Yoyogi-5-chome, Shibuya-ku Tokyo, (JP)
Maezawa, Hiroyuki, 11-5-1, Toyogaoka-2-chome, Tama-shi, (JP)
Yamaguchi, Junko, 236, Miwamachi, Machida-shi, (JP)
Katumata, Hidetosi, 28-2, Sagamigaoka-2-chome, Zama-shi, (JP)
Oshio, Takeshi, 8-1, Yanagibashi-1-chome, Yamato-shi, (JP)

LEGAL REPRESENTATIVE:

Patentanwalte Beetz - Timpe - Siegfried Schmitt-Fumian - Mayr (100712),
Steinsdorfstrasse 10, D-80538 Munchen, (DE)

PATENT (CC, No, Kind, Date): EP 268293 A2 880525 (Basic)
EP 268293 A3 920513
EP 268293 B1 951018

APPLICATION (CC, No, Date): EP 87117073 871119;

PRIORITY (CC, No, Date): JP 86275207 861120

DESIGNATED STATES: DE

INTERNATIONAL PATENT CLASS: G06F-009/44 ; G06F-011/00

CITED PATENTS (EP A): US 3711863 A

CITED REFERENCES (EP A):

INF. PROCESSING 86, PROC. OF THE IFIP 10TH WORLD COMPUTER CONGRESS
September 1986, DUBLIN, IRELAND pages 675 - 684; H. EHRIG ET AL.:
'Programming in the large with algebraic module specifications'
PROC. IEEE/AIAA 7TH DIGITAL AVIONICS SYSTEMS CONF. 13 October 1986,
TEXAS, US pages 33 - 40; J.H.SCHWARTZ: 'The bendix computer -aided
software engineering system: a new approach to an ada* design language'
COMPUTER. vol. 18, no. 8, August 1985, LONG BEACH US pages 27 - 35;
G.P.BROWN ET AL.: 'Program visualization: graphical support for
software development'
IN: SIGPLAN NOTICES 1984 vol. 19, no. 5, 1984, pages 30 - 41; S.P. REISS:
'Graphical program development with pecan program development systems'
IEEE MICRO. vol. 6, no. 3, June 1986, NEW YORK US pages 26 - 33;
P.CORSINI ET AL: 'Multibug: interactive debugging in distributed
systems';

ABSTRACT EP 268293 A2

A display method in an information processing device for use in software development support comprises the steps of analyzing a software product before change and a software product after change, respectively to form structure information before change and structure information after change, which consist of software element informations correlated with each other, in accordance with the respective structures of the software products; forming structure information that forms a union set of the before-change structure information and said afterchange structure information, which includes all the software element informations before and after change, and appending identification information representative of the manner of change in each software element to the union set structure information; and converting at least a part of the union set structure information into its graphic format so as to be produced together with the representations of the change manners represented by the identification information. (see image in original document)

ABSTRACT WORD COUNT: 154

LEGAL STATUS (Type, Pub Date, Kind, Text):

Application: 880525 A2 Published application (A1with Search Report
;A2without Search Report)
Examination: 910109 A2 Date of filing of request for examination:
901113
Search Report: 920513 A3 Separate publication of the European or
International search report
Examination: 931201 A2 Date of despatch of first examination report:
931013
Change: 940316 A2 Representative (change)
Grant: 951018 B1 Granted patent
Oppn None: 961009 B1 No opposition filed

LANGUAGE (Publication,Procedural,Application): English; English; English

FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	622
CLAIMS B	(English)	EPAB95	203
CLAIMS B	(German)	EPAB95	202
CLAIMS B	(French)	EPAB95	262
SPEC A	(English)	EPABF1	5080
SPEC B	(English)	EPAB95	5191

Total word count - document A 5702
Total word count - document B 5858
Total word count - documents A + B 11560

INTERNATIONAL PATENT CLASS: G06F-009/44 ...

...ABSTRACT information; and converting at least a part of the union set structure information into its **graphic** format so as to be produced together with the representations of the **change** manners represented by the identification information. (see image in original document)

...SPECIFICATION between an old program and a new program must be made for checking if the **changes** have been made as desired. In the prior art software development support system, the **comparison** is made, for each line in the text, between the **program** before **change** and the **program** after **change** to check the **changes** (addition, deletion and **updating**). The results of the **comparison** are displayed or printed out in the form of texts with identifiers respectively indicative of...

...Thus, the first object of the present invention is to provide a display method that **graphically** displays the positions and types of the **changes** of a program, thereby intuitively and synthetically understanding the change to improve the efficiency and...

...at least a part of the structure information that forms a union set, into a **graphic** form to output the **graphic** and the **change** represented by said identifier.

A second feature of the present invention resides in comprising the...

...information includes all the information of the software elements not changed and changed (addition, deletion, **updating**) and **software** structure information. The information indicative of the manner of change (presence and ...change. On the basis of the thus obtained union-set-forming-structure information with the **change** -indicating identifier, the display of software in the **graphic** form and the manner of the **change** of the software elements on the **graphic** display can be made, so that the contents of **change** of the software can be easily understood.

In accordance with the second feature of the...

Set, Items Description
S1 6467038 PATCH? OR UPDAT? OR UPGRAD? OR UP() (DAT? OR GRAD?) OR REV-
S? OR REPAIR? OR FIX OR FIXING OR FIXED
S2 16517519 APPLICATION? OR PROGRAM? OR SOFTWARE? OR APP OR MODULE? OR
BINARY OR EXECUTABLE() CODE?
S3 13649727 DELTA OR DELTAS OR CHANG? OR DIFFER? OR VARIATION? OR VARI-
ANC? OR VARY?
S4 3421941 CFG OR GRAPH? OR FLOWGRAPH? OR DIAGRAM? OR CHART?
S5 35243 (NEIGHBOR? OR NEIGHBOUR? OR ADJACENT? OR NEAR? OR CONTIGUO-
US? OR SURROUND?) (5N) (BLOCK? ? OR NODE? OR MODULE?)
S6 9607271 IDENTIF? OR COMPAR? OR JUDGE? OR JUDGING OR CONTRAST? OR M-
ATCH? OR UNMATCH?
S7 9449223 IDENTICAL? OR SIMILAR? OR SAME? OR EQUIVALENT?
S8 9010744 SOURCE? OR TARGET?
S9 3 S1(3N)S2(S)S3(S)S4(S)S5
S10 1275 (S3 OR S4) (3N) (S6 OR S7) (S)S1(3N)S2
S11 484 (S8 OR S5) AND S10
S12 21 S10(5N) (S8 OR S5)
S13 24 S9 OR S12
S14 20 RD (unique items)
S15 17 S14 NOT PY>2001
S16 16 S15 NOT PD>20011114
File 275:Gale Group Computer DB(TM) 1983-2004/Jan 26
(c) 2004 The Gale Group
File 47:Gale Group Magazine DB(TM) 1959-2004/Jan 26
(c) 2004 The Gale group
File 75:TGG Management Contents(R) 86-2004/Jan W3
(c) 2004 The Gale Group
File 636:Gale Group Newsletter DB(TM) 1987-2004/Jan 26
(c) 2004 The Gale Group
File 16:Gale Group PROMT(R) 1990-2004/Jan 26
(c) 2004 The Gale Group
File 624:McGraw-Hill Publications 1985-2004/Jan 26
(c) 2004 McGraw-Hill Co. Inc
File 484:Periodical Abs Plustext 1986-2004/Jan W3
(c) 2004 ProQuest
File 613:PR Newswire 1999-2004/Jan 27
(c) 2004 PR Newswire Association Inc
File 813:PR Newswire 1987-1999/Apr 30
(c) 1999 PR Newswire Association Inc
File 141:Readers Guide 1983-2004/Dec
(c) 2004 The HW Wilson Co
File 621:Gale Group New Prod.Annou.(R) 1985-2004/Jan 26
(c) 2004 The Gale Group
File 674:Computer News Fulltext 1989-2004/Jan W4
(c) 2004 IDG Communications
File 160:Gale Group PROMT(R) 1972-1989
(c) 1999 The Gale Group
File 15:ABI/Inform(R) 1971-2004/Jan 27
(c) 2004 ProQuest Info&Learning
File 9:Business & Industry(R) Jul/1994-2004/Jan 26
(c) 2004 Resp. DB Svcs.
File 13:BAMP 2004/Jan W2
(c) 2004 Resp. DB Svcs.
File 810:Business Wire 1986-1999/Feb 28
(c) 1999 Business Wire
File 610:Business Wire 1999-2004/Jan 27
(c) 2004 Business Wire.
File 647:CMP Computer Fulltext 1988-2004/Jan W3
(c) 2004 CMP Media, LLC
File 148:Gale Group Trade & Industry DB 1976-2004/Jan 26
(c) 2004 The Gale Group
File 634:San Jose Mercury Jun 1985-2004/Jan 26
(c) 2004 San Jose Mercury News

16/3,K/3 (Item 3 from file: 275)
DIALOG(R)File 275:Gale Group Computer DB(TM)
(c) 2004 The Gale Group. All rts. reserv.

01293632 SUPPLIER NUMBER: 07169168 (USE FORMAT 7 OR 9 FOR FULL TEXT)
Automating code generation. (Tools of the Trade) (column)
Keuffel, Warren
Computer Language, v6, n4, p25(6)
April, 1989
DOCUMENT TYPE: column ISSN: 0749-2839 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 2065 LINE COUNT: 00161

... own code.

Third, reverse engineering of code to specifications will become a useful tool for **programmers**. Need to **fix** a bug? Just run your **source** code through the analyzer and debug your code graphically. Identify a module in the structure...

16/3,K/13 (Item 1 from file: 674)
DIALOG(R)File 674:Computer News Fulltext
(c) 2004 IDG Communications. All rts. reserv.

080811

King of the NOS hill

NetWare holds the performance reins, but Windows 2000 reigns supreme for features overall.

Byline: JOHN BASS AND JAMES ROBINSON, NETWORK WORLD TEST ALLIANCE

Journal: Network World Page Number: 71

Publication Date: January 24, 2000

Word Count: 4698 Line Count: 427

Text:

...is that there is a NOS waiting just for you. After the rash of recent **software revisions**, we took an in-depth look at four of the major NOSes on the market...

... flag is disabled, NetWare writes to disk in a more efficient manner by batching together **contiguous blocks** of data on the cache and writing all those blocks to disk at once. Likewise...is the glue that holds most of the Windows 2000 management functionality together. This configurable **graphical** user interface (GUI) lets you snap in Microsoft and third-party applets that customize its...

...directory service. From the Active Directory management tool inside MMC, you can configure users and **change** policies. The network configuration tools are found in a separate application that opens when you...

...on the desktop. Each network interface is listed inside this window. You can add and **change** protocols and configure, enable and disable interfaces from here without rebooting. NetWare offers several interfaces...

... of tools for server configuration. One of the most useful is NWConfig, which lets you **change** start-up files, install system modules and configure the storage subsystem. NWConfig is simple, intuitive and predictable. ConsoleOne is a Java-based interface with a few **graphical** tools for managing and configuring NetWare. Third-party administration tools can plug into ConsoleOne and...

... Red Hat's overall systems management interface is called LinuxConf and can run as a **graphical** or text-based application. The **graphical** interface, which resembles that of MMC, works well but has some layout issues that make...

... to files residing on a Linux server - and FTP and Web servers. You can apply **changes** without rebooting the system. Overall, Red Hat's interface is useful and the underlying tools...

... while others are text-based. The server required a reboot to apply many of the **changes**. On the plus side, you can manage multiple UnixWare servers from SCOAdmin. SCO also offers...

... server's internals. The Windows 2000 System Monitor lets you view a real-time, running **graph** of system operations, such as CPU and network utilization, and memory and disk usage. We...

...command-line tools for monitoring the server, such as iostat and vmstat. It has no **graphical** monitoring tools. As with any Unix operating system, you can write scripts to automate these...

... typically cryptic and require a high level of proficiency to use effectively. A suite of **graphical** monitoring tools would be a great addition to Red Hat's Linux distribution. UnixWare also...remotely set file and directory permissions from a Windows client, as well as create and **change** users and their settings. SCO and Red Hat offer support for the Unix-based Network...

... to configure and use. Storage management Windows 2000 provides the best

tools for storage management. Its **graphical** Manage Disks tool for local disk configuration includes software RAID management; you can dynamically add...

... give it high marks for functionality and ease of use. Red Hat Linux offers no **graphical** RAID configuration tools, but its command line tools made RAID configuration easy. To configure disks on the UnixWare server, we used the Veritas Volume Manager **graphical** disk and volume administration tool that ships with UnixWare. We had some problems initially getting...

... UnixWare has a set of security tools called Security Manager that lets you set up **varying** degrees of intrusion protection across your network services, from no restriction to turning all network...restore facility. Backups can be all-inclusive, cover parts of a volume or store a **differential** snapshot. Red Hat provides a load-balancing product called piranha with its Linux. This package...preliminary testing to establish the test parameters, then ran those parameters against each NOS. We **graphed** the results of each file test on a curve with five data points. The curves
...

... test the network performance of each NOS, we used Ganymede Software's Chariot software, which **differs** from the Benchmark Factory software in that all file transactions occur in memory. The disk...

Set	Items	Description
S1	14595	PATCH? OR UPDAT? OR UPGRAD? OR UP() (DAT? OR GRAD?) OR REVI- S? OR REPAIR? OR FIX OR FIXING OR FIXED
S2	84819	APPLICATION? OR PROGRAM? OR SOFTWARE? OR APP OR MODULE? OR BINARY OR EXECUTABLE() CODE?
S3	21016	DELTA OR DELTAS OR CHANG? OR DIFFER? OR VARIATION? OR VARI- ANC? OR VARY?
S4	21328	CFG OR GRAPH? OR FLOWGRAPH? OR DIAGRAM? OR CHART?
S5	17	(NEIGHBOR? OR NEIGHBOUR? OR ADJACENT? OR NEAR? OR CONTIGUO- US? OR SURROUND?) (5N) (BLOCK? ? OR NODE? OR MODULE?)
S6	11705	IDENTIF? OR COMPAR? OR JUDGE? OR JUDGING OR CONTRAST? OR M- ATCH? OR UNMATCH?
S7	12256	IDENTICAL? OR SIMILAR? OR SAME? OR EQUIVALENT?
S8	11665	SOURCE? OR TARGET?
S9	0	S1 AND S2 AND S3 AND S4 AND S5
S10	23	(S3 OR S4) (3N) (S6 OR S7) AND S1 (3N) S2
S11	2	S10 AND (S8 OR S5)

File 256:SoftBase:Reviews,Companies&Prods. 82-2004/Dec

(c) 2004 Info.Sources Inc

11/3, K/1
DIALOG(R) File 256:SoftBase:Reviews,Companies&Prods.
(c) 2004 Info.Sources Inc. All rts. reserv.

01139033 DOCUMENT TYPE: Product

PRODUCT NAME: PowerGEM Plus (139033)

Fujitsu Software Technology Corp (Fujitsu-Softek (394572)
1250 E Arques Ave M/S 317
Sunnyvale, CA 94088-3470 United States
TELEPHONE: (408) 746-7638

RECORD TYPE: Directory

CONTACT: Sales Department

REVISION DATE: 20030316

...set up local and network repositories; manage file check-in and check-out processes; view **change** histories; and **compare source** versions. PowerGEM Plus provides multiple users with simultaneous access to resources. However, the system restricts...

...version and history statistics. PowerGEM Plus's Development Manager supports the reviewing and testing of **application updates**.

11/3, K/2

DIALOG(R) File 256:SoftBase:Reviews,Companies&Prods.
(c) 2004 Info.Sources Inc. All rts. reserv.

01062731 DOCUMENT TYPE: Product

PRODUCT NAME: HTML Compare (062731)

Applian Technologies Inc (650226)
24 7th Ave
San Francisco, CA 94118 United States
TELEPHONE: (415) 831-1499

RECORD TYPE: Directory

CONTACT: Sales Department

REVISION DATE: 20011130

...to track HTML page and Web site edits. Operating something like a word processor's **revision tracking module** or a 'diff' utility, HTML Compare highlights **differences** between Web page **revisions**. The **program** identifies additions, deletions, and other modifications. Tapping an intuitive interface, users can employ the program...
...edits that have been made to multiple pages. As well, developers can purchase HTML Compare **source code**, allowing them to add the program's features to other applications.

Set	Items	Description
S1	30	AU=(VENKATESAN, R OR VENKATESAN R)
S2	77	AU=(SINHA, S? OR SINHA S?)
S3	3	S1 AND S2
S4	2	(S1 OR S2) AND IC=G06F-009?
S5	24	(S1 OR S2) AND IC=G06F?
S6	26	S3 OR S4 OR S5
S7	26	IDPAT (sorted in duplicate/non-duplicate order)
S8	24	IDPAT (primary/non-duplicate records only)
File 344:Chinese Patents Abs Aug 1985-2003/Nov		
		(c) 2003 European Patent Office
File 347:JAPIO Oct 1976-2003/Sep(Updated 040105)		
		(c) 2004 JPO & JAPIO
File 348:EUROPEAN PATENTS 1978-2004/Jan W04		
		(c) 2004 European Patent Office
File 349:PCT FULLTEXT 1979-2002/UB=20040122,UT=20040115		
		(c) 2004 WIPO/Univentio
File 350:Derwent WPIX 1963-2004/UD,UM &UP=200406		
		(c) 2004 Thomson Derwent

8/5/7 (Item 7 from file: 350)
DIALOG(R) File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

014652361 **Image available**
WPI Acc No: 2002-473065/200251
XRPX Acc No: N02-373459

Delta generation method for program binaries, involves identifying unmatched blocks which are merged into source control flow graph representation so that source and target Control Flow Graph's (CFG's) are identical

Patent Assignee: MICROSOFT CORP (MICKT)

Inventor: SINHA S ; VENKATESAN R

Number of Countries: 027 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 1205842	A2	20020515	EP 2001126979	A	20011113	200251 B
JP 2002169702	A	20020614	JP 2001349299	A	20011114	200254

Priority Applications (No Type Date): US 2000713633 A 20001114

Patent Details:

Patent No Kind Lan Pg Main IPC Filing Notes

EP 1205842 A2 E 29 G06F-009/44

Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT
LI LT LU LV MC MK NL PT RO SE SI TR

JP 2002169702 A 23 G06F-011/00

Abstract (Basic): EP 1205842 A2

NOVELTY - Control flow graph (CFG) representations of a source program (112) and a target program (122) are compared to identify matched and unmatched blocks. Edit operations that merge the unmatched blocks into the source representation, is determined so that source and target representations are identical. A delta (142) comprising the unmatched blocks and edit operations, is produced.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for :

- (1) Computer readable medium for storing delta generation program;
- (2) Block matching method;
- (3) Patch data structure;
- (4) Patch data structure transmission method;
- (5) Source program patching method;
- (6) Delta generator system;
- (7) Computer readable medium storing data structure comprising generated delta;
- (8) Computer readable medium storing block matching program;
- (9) Method for matching procedures between CFG representations of the portions of programs;
- (10) Computer readable medium storing CFG portion matching procedure program;
- (11) Method for facilitating matching of blocks between CFG representations of the programs;
- (12) Computer readable medium storing block matching facilitating program;
- (13) Computer readable medium storing data structure comprising delta produced by delta generator system.

USE - For generating delta between program binaries.

ADVANTAGE - Common blocks of source and targets CFG's are matched in multiple passes so as to improve the matching by relaxing the criteria for a match and the register renaming problems is solved so that blocks can be fairly compared.

DESCRIPTION OF DRAWING(S) - The figure shows the schematic block diagram of the minimum delta generator for program binaries.

Source program (112)

Target program (122)

Delta (142)

pp; 29 DwgNo 1/9

Title Terms: DELTA; GENERATE; METHOD; PROGRAM; BINARY; IDENTIFY; UNMATCHED;

BLOCK; MERGE; SOURCE; CONTROL; FLOW; GRAPH; REPRESENT; SO; SOURCE; TARGET
; CONTROL; FLOW; GRAPH; IDENTICAL

Derwent Class: T01

International Patent Class (Main): G06F-009/44 ; G06F-011/00

File Segment: EPI

8/5/8 (Item 8 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

014615009 **Image available**
WPI Acc No: 2002-435713/200246

XRPX Acc No: N02-342963

Image hashing by deriving independent hash values for visually distinct images and identical values for similar images

Patent Assignee: MICROSOFT CORP (MICKT)

Inventor: KOON S W; VENKATESAN R

Number of Countries: 094 Number of Patents: 003

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
WO 200237331	A1	20020510	WO 2000US41359	A	20001019	200246 B
AU 200245857	A	20020515	WO 2000US41359	A	20001019	200258
			AU 200245857	A	20001019	
EP 1327201	A1	20030716	EP 2000993908	A	20001019	200347
			WO 2000US41359	A	20001019	

Priority Applications (No Type Date): WO 2000US41359 A 20001019

Patent Details:

Patent No Kind Lan Pg Main IPC Filing Notes

WO 200237331 A1 E 30 G06F-017/30

Designated States (National): AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CR CU CZ DE DK DM DZ EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG UZ VN YU ZA ZW

Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR IE IT KE LS LU MC MW MZ NL OA PT SD SE SL SZ TZ UG ZW

AU 200245857 A G06F-017/30 Based on patent WO 200237331

EP 1327201 A1 E G06F-017/30 Based on patent WO 200237331

Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI

Abstract (Basic): WO 200237331 A1

NOVELTY - Method consists in deriving a hash value representing the received image so that visually distinct images result in hash values that are approximately independent of each other and visually similar images result in identical hash values. The hash value is stored with the image to index it and watermark the image. Hash values from different images are compared.

DETAILED DESCRIPTION - There are INDEPENDENT CLAIMS for (1) a digital image processing system, (2) a digital image hash system, (3) a hash value program.

USE - Method is for hashing digital images in databases and can be used for on-line searches of web sites for detection of pirated copies..

ADVANTAGE - Method allows modest changes to an image which may or may not be perceptible to the eye without resulting in different hash values for the original and modified images.

DESCRIPTION OF DRAWING(S) - The figure shows a block diagram of an image distribution system.

pp; 30 DwgNo 1/6

Title Terms: IMAGE; HASH; DERIVATIVE; INDEPENDENT; HASH; VALUE; VISUAL;

DISTINCT; IMAGE; IDENTICAL; VALUE; SIMILAR; IMAGE

Derwent Class: T01

International Patent Class (Main): G06F-017/30

International Patent Class (Additional): G06T-001/00

File Segment: EPI

Set	Items	Description
S1	1910775	PATCH? OR UPDAT? OR UPGRAD? OR UP() (DAT? OR GRAD?) OR REV- S? OR REPAIR? OR FIX OR FIXING OR FIXED
S2	1467018	APPLICATION? OR PROGRAM? OR SOFTWARE? OR APP OR MODULE? OR BINARY OR EXECUTABLE() CODE?
S3	3049393	DELTA OR DELTAS OR CHANG? OR DIFFER? OR VARIATION? OR VARI- ANC? OR VARY?
S4	892321	CFG OR GRAPH? OR FLOWGRAPH? OR DIAGRAM? OR CHART?
S5	18398	(NEIGHBOR? OR NEIGHBOUR? OR ADJACENT? OR NEAR? OR CONTIGUO- US? OR SURROUND?) (5N) (BLOCK? ? OR NODE? OR MODULE?)
S6	1410605	IDENTIF? OR COMPAR? OR CONTRAST? OR MATCH? OR UNMATCH?
S7	1864356	IDENTICAL? OR SIMILAR? OR SAME? OR EQUIVALENT?
S8	1261760	SOURCE? OR TARGET?
S9	3250	S1 AND S2 AND S3 AND S4
S10	2	S9 AND S5 AND (S6 OR S7)
S11	530	S1(3N)S2 AND S3 AND S4
S12	142	S11 AND (S6 OR S7)
S13	41	S11 AND S8
S14	43	S10 OR S13
S15	26	S14 AND IC=G06F?
S16	26	IDPAT (sorted in duplicate/non-duplicate order)
S17	25	IDPAT (primary/non-duplicate records only)
File 347:JAPIO Oct 1976-2003/Sep(Updated 040105)		
(c) 2004 JPO & JAPIO		
File 350:Derwent WPIX 1963-2004/UD,UM &UP=200406		
(c) 2004 Thomson Derwent		

17/5/11 (Item 11 from file: 350)

DIALOG(R) File 350:Derwent WPIX

(c) 2004 Thomson Derwent. All rts. reserv.

014652361 **Image available**

WPI Acc No: 2002-473065/200251

XRPX Acc No: N02-373459

Delta generation method for program binaries, involves identifying unmatched blocks which are merged into source control flow graph representation so that source and target Control Flow Graph 's (CFG 's) are identical

Patent Assignee: MICROSOFT CORP (MICKT)

Inventor: SINHA S; VENKATESAN R

Number of Countries: 027 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
EP 1205842	A2	20020515	EP 2001126979	A	20011113	200251 B
JP 2002169702	A	20020614	JP 2001349299	A	20011114	200254

Priority Applications (No Type Date): US 2000713633 A 20001114

Patent Details:

Patent No Kind Lan Pg Main IPC Filing Notes

EP 1205842 A2 E 29 G06F-009/44

Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT
LI LT LU LV MC MK NL PT RO SE SI TR

JP 2002169702 A 23 G06F-011/00

Abstract (Basic): EP 1205842 A2

NOVELTY - Control flow graph (CFG) representations of a **source** program (112) and a **target** program (122) are compared to identify matched and unmatched blocks. Edit operations that merge the unmatched blocks into the **source** representation, is determined so that **source** and **target** representations are identical. A **delta** (142) comprising the unmatched blocks and edit operations, is produced.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for :

- (1) Computer readable medium for storing **delta** generation program;
- (2) Block matching method;
- (3) Patch data structure;
- (4) Patch data structure transmission method;
- (5) **Source program patching** method;
- (6) **Delta** generator system;
- (7) Computer readable medium storing data structure comprising generated **delta** ;
- (8) Computer readable medium storing block matching program;
- (9) Method for matching procedures between **CFG** representations of the portions of programs;
- (10) Computer readable medium storing **CFG** portion matching procedure program;
- (11) Method for facilitating matching of blocks between **CFG** representations of the programs;
- (12) Computer readable medium storing block matching facilitating program;
- (13) Computer readable medium storing data structure comprising **delta** produced by **delta** generator system.

USE - For generating **delta** between program binaries.

ADVANTAGE - Common blocks of **source** and **targets** **CFG** 's are matched in multiple passes so as to improve the matching by relaxing the criteria for a match and the register renaming problems is solved so that blocks can be fairly compared.

DESCRIPTION OF DRAWING(S) - The figure shows the schematic block diagram of the minimum **delta** generator for program binaries.

Source program (112)

Target program (122)

Delta (142)

pp; 29 DwgNo 1/9

Title Terms: **DELTA** ; GENERATE; METHOD; PROGRAM; BINARY; IDENTIFY; UNMATCHED; BLOCK; MERGE; **SOURCE** ; CONTROL; FLOW; **GRAPH** ; REPRESENT; SO;

SOURCE ; TARGET ; CONTROL; FLOW; GRAPH ; IDENTICAL
Derwent Class: T01
International Patent Class (Main): G06F-009/44 ; G06F-011/00
File Segment: EPI

17/5/13 (Item 13 from file: 350)

DIALOG(R)File 350:Derwent WPIX

(c) 2004 Thomson Derwent. All rts. reserv.

013956946 **Image available**

WPI Acc No: 2001-441160/200147

Related WPI Acc No: 2001-354652; 2001-397419; 2001-456994; 2001-457005;
2001-464784; 2001-580592; 2001-596340

XRPX Acc No: N01-326395

Network system for content collaboration among group of participants; uses logic in communication with database to asynchronously dynamically update binary content in dynamic content region in response to input

Patent Assignee: FIREDROP INC (FIRE-N); ZAPLET INC (ZAPL-N)

Inventor: AXE B; EVANS S R; HANSON M; MILLER G

Number of Countries: 094 Number of Patents: 003

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
WO 200122246	A1	20010329	WO 2000US40745	A	20000824	200147 B
AU 200126127	A	20010424	AU 200126127	A	20000824	200147
US 6507865	B1	20030114	US 99151476	P	19990830	200313
			US 99151650	P	19990831	
			US 99426648	A	19991025	
			US 99427152	A	19991025	
			US 99427378	A	19991025	
			US 2000483221	A	20000114	

Priority Applications (No Type Date): US 2000483221 A 20000114; US 99151476 P 19990830; US 99151650 P 19990831; US 99426648 A 19991025; US 99427152 A 19991025; US 99427378 A 19991025

Patent Details:

Patent No Kind Lan Pg Main IPC Filing Notes

WO 200122246 A1 E 54 G06F-015/16

Designated States (National): AE AG AL AM AT AU AZ BA BB BG BR BY BZ CA CH CN CR CU CZ DE DK DM DZ EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG US UZ VN YU ZA ZW
Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR IE IT KE LS LU MC MW MZ NL OA PT SD SE SL SZ TZ UG ZW

AU 200126127 A G06F-015/16 Based on patent WO 200122246

US 6507865 B1 G06F-015/16 Provisional application US 99151476

Provisional application US 99151650

CIP of application US 99426648

CIP of application US 99427152

CIP of application US 99427378

Abstract (Basic): WO 200122246 A1

NOVELTY - At least one dynamic content region in an electronic medium has binary content. An interface region in the electronic medium accepts input from one of any of the participants and an external source in data communication with a server. Logic is in communication with the database to asynchronously dynamically update the binary content in the dynamic content region in response to the input.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for:

(a) a method of content collaboration among a group of participants

(b) a content collaboration tool

(c) a method for creating a greeting card among group of participants

(d) a method for managing tasks among group of participants

(e) a method of tracking stocks among group of participants

(f) a network system

(g) a computer software residing on a computer readable medium at device connected to network

USE - In content collaboration among a group of participants connected to networks using a dynamic distribution of data

ADVANTAGE - Improves access to content that may be checked out, modified, and then checked back into some repository. Reduces the time required for each participant to make his or her changes excluding

problem of locking-unlocking of the content or keep checking to see if the content is unlocked.

DESCRIPTION OF DRAWING(S) - The drawing is a **diagram** of a data structure for a media for communicating information and supports collaboration among participants in group connected to network (referred as a 'zaplet').

pp; 54 DwgNo 4/17

Title Terms: NETWORK; SYSTEM; CONTENT; GROUP; PARTICIPATING; LOGIC; COMMUNICATE; DATABASE; ASYNCHRONOUS; DYNAMIC; UPDATE; BINARY; CONTENT; DYNAMIC; CONTENT; REGION; RESPOND; INPUT

Derwent Class: T01

International Patent Class (Main): **G06F-015/16**

File Segment: EPI

17/5/14 (Item 14 from file: 350)

DIALOG(R)File 350:Derwent WPIX

(c) 2004 Thomson Derwent. All rts. reserv.

013612162 **Image available**

WPI Acc No: 2001-096370/200111

XRPX Acc No: N01-073196

Automatic patch generation apparatus for software in high language, has load- module comparison patch data generator which processes patch code section for patch revision and unconditional branch instruction

Patent Assignee: HITACHI COMPUTER ELECTRONICS KK (HITA-N); HITACHI LTD (HITA)

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
JP 2000330780	A	20001130	JP 99138244	A	19990519	200111 B

Priority Applications (No Type Date): JP 99138244 A 19990519

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 2000330780	A	9		G06F-009/06	

Abstract (Basic): JP 2000330780 A

NOVELTY - An unconditional branch instruction path unit (6) performs the patch of the instruction code of the lead of a function before modification on a corrected load module to an unconditional branch instruction at the lead of the function after modification. A load-module comparison path data generator (7) processes only the patch code section for patch revision and the unconditional branch instruction as patch data.

DETAILED DESCRIPTION - A **source** file correction device (1) performs the correction of a master code. A **source** comparison and recording device (2) compares the master **source** and the corrected **source**, and records the comparison result to a file per function. A synthetic **source** generator (3) produces a synthetic **source** which provides a full-static variable **change** and addition into an amendment static variable section for path revision. A synthetic **source** object generator (4) produces and compiles a synthetic **source** object. A corrected load module generator (5) produces a corrected load module and a corrected static variable area.

USE - For producing **patch** data opposing to **software** developed in high language.

ADVANTAGE - Enables **source** correction in high language and surface correction of link with identical production and quality, since automatic generation of patch data is performed from a **source** file. Execution order of instruction of CPU can be generated automatically when a compiler generates **different** instruction execution orders. Ensures generation of exact **patch** opposing to **software** in high language.

DESCRIPTION OF DRAWING(S) - The figure shows the block **diagram** of automatic patch generation apparatus.

Source file correction device (1)

Source comparison and recording device (2)

Synthetic **source** generator (3)

Synthetic **source** object generator (4)

Corrected load module generator (5)

Unconditional branch instruction path unit (6)

Load-module comparison path data generator (7)

pp; 9 DwgNo 1/3

Title Terms: AUTOMATIC; PATCH; GENERATE; APPARATUS; SOFTWARE; HIGH; LANGUAGE; LOAD; MODULE; COMPARE; PATCH; DATA; GENERATOR; PROCESS; PATCH; CODE; SECTION; PATCH; REVISED; UNCONDITIONAL; BRANCH; INSTRUCTION

Derwent Class: T01

International Patent Class (Main): G06F-009/06

File Segment: EPI

i7/5/24 (Item 24 from file: 347)
DIALOG(R)File 347:JAPIO
(c) 2004 JPO & JAPIO. All rts. reserv.

06736501 **Image available**
REMOTE MAINTENANCE SYSTEM

PUB. NO.: 2000-322348 [JP 2000322348 A]
PUBLISHED: November 24, 2000 (20001124)
INVENTOR(s): YAMAMOTO SHUJI
 TAKEHANA YOICHI
APPLICANT(s): YOKOGAWA ELECTRIC CORP
APPL. NO.: 11-133780 [JP 99133780]
FILED: May 14, 1999 (19990514)
INTL CLASS: G06F-013/00

ABSTRACT

PROBLEM TO BE SOLVED: To realize a remote maintenance system flexibly adaptable to the **change** of **target** equipment, maintenance function, and protocol.

SOLUTION: In a remote maintenance system for executing prescribed maintenance by communicating with **target** equipment from a remote place, the corresponding **chart** of a **target** control command for instructing the control of **target** equipment 8 to a **target** control program name for controlling the **target** equipment 8 and a **target** control program collection being the set of **target** control programs are prepared, and the corresponding **chart** and the **target** control **program** collection are **updated** as necessary.

COPYRIGHT: (C)2000, JPO

Set Items Description
S1 3935 AU=(SINHA S? OR SINHA, S?)
S2 490 AU=(VENKATESAN R? OR VENKATESAN, R?)
S3 4 S1 AND S2
S4 1201 (S1 OR S2) AND (DELTA? OR INCONSISTEN? OR CHANG? OR VARIATION? OR DIFFEREN?)
S5 152 S4 AND (COMPUTER() (PROGRAM? OR SYSTEM?) OR APPLICATION? OR APP OR SOFTWARE?)
S6 0 S5 AND (NEIGHBORHOOD? OR NEIGHBOURHOOD? OR (SURROUND? OR N-EARBY OR ADJACENT? OR NEAR OR CONTIGUOUS) (N)BLOCK? ? OR CFG OR CONTROL() FLOW OR FLOW()GRAPH?)
S7 2 S5 AND (PATCH? OR UPGRAD? OR UPDAT? OR UP() (GRADE OR GRADING OR GRADES OR GRADED OR DATE OR DATES OR DATING OR DATED))
S8 6 S3 OR S7
S9 5 RD (unique items)
File 2:INSPEC 1969-2004/Jan W3
 (c) 2004 Institution of Electrical Engineers
File 6:NTIS 1964-2004/Jan W4
 (c) 2004 NTIS, Intl Cpyrght All Rights Res
File 8:Ei Compendex(R) 1970-2004/Jan W3
 (c) 2004 Elsevier Eng. Info. Inc.
File 35:Dissertation Abs Online 1861-2004/Dec
 (c) 2004 ProQuest Info&Learning
File 34:SciSearch(R) Cited Ref Sci 1990-2004/Jan W3
 (c) 2004 Inst for Sci Info
File 148:Gale Group Trade & Industry DB 1976-2004/Jan 26
 (c)2004 The Gale Group
File 65:Inside Conferences 1993-2004/Jan W4
 (c) 2004 BLDSC all rts. reserv.
File 94:JICST-EPlus 1985-2004/Jan W3
 (c)2004 Japan Science and Tech Corp(JST)
File 111:TGG Natl.Newspaper Index(SM) 1979-2004/Jan 26
 (c) 2004 The Gale Group
File 275:Gale Group Computer DB(TM) 1983-2004/Jan 26
 (c) 2004 The Gale Group
File 647:cmp Computer Fulltext 1988-2004/Jan W3
 (c) 2004 CMP Media, LLC
File 674:Computer News Fulltext 1989-2004/Jan W3
 (c) 2004 IDG Communications
File 636:Gale Group Newsletter DB(TM) 1987-2004/Jan 26
 (c) 2004 The Gale Group

9/5,K/1 (Item 1 from file: 2)
DIALOG(R)File 2:INSPEC

(c) 2004 Institution of Electrical Engineers. All rts. reserv.

7726044 INSPEC Abstract Number: C2003-10-6130S-076

Title: **Oblivious hashing: a stealthy software integrity verification primitive**

Author(s): Yuqun Chen; **Venkatesan, R.** ; Cary, M.; Ruoming Pang; **Sinha, S.** ; Jakubowski, M.H.

Author Affiliation: Microsoft Res., Redmond, WA, USA

Conference Title: Information Hiding. 5th International Workshop, IH 2002. Revised Papers (Lecture Notes in Computer Science Vol.2578) p. 400-14

Editor(s): Petitcolas, F.A.P.

Publisher: Springer Verlag, Berlin, Germany

Publication Date: 2002 Country of Publication: Germany vi+426 pp.

ISBN: 3 540 00421 1 Material Identity Number: XX-2003-00281

Conference Title: Information Hiding. 5th International Workshop, IH 2002. Revised Papers

Conference Date: 7-9 Oct. 2002 Conference Location: Noordwijk, Netherlands

Language: English Document Type: Conference Paper (PA)

Treatment: Applications (A); Practical (P)

Abstract: We describe a novel software verification primitive called Oblivious Hashing. Unlike previous techniques that mainly verify the static shape of code, this primitive allows implicit computation of a hash value based on the actual execution (i.e., space-time history of computation) of the code. We also discuss its applications in local software tamper resistance and remote code authentication. (20 Refs)

Subfile: C

Descriptors: authorisation; cryptography; data integrity; program diagnostics; program verification

Identifiers: stealthy software integrity verification primitive; Oblivious Hashing; hash value; local software tamper resistance; remote code authentication

Class Codes: C6130S (Data security); C6110F (Formal methods); C6150G (Diagnostic, testing, debugging and evaluating systems)

Copyright 2003, IEE

Author(s): Yuqun Chen; **Venkatesan, R.** ; Cary, M.; Ruoming Pang; **Sinha, S.** ; Jakubowski, M.H.

9/5,K/2 (Item 1 from file: 34)

DIALOG(R)File 34:SciSearch(R) Cited Ref Sci

(c) 2004 Inst for Sci Info. All rts. reserv.

11508743 Genuine Article#: BW39Y Number of References: 18

Title: **Oblivious hashing: A stealthy software integrity verification primitive**

Author(s): Chen YQ (REPRINT) ; **Venkatesan R** ; Cary M; Pang RM; **Sinha S** ; Jakubowski MH

Corporate Source: Microsoft Corp, Res, 1 Microsoft Way/Redmond//WA/98052 (REPRINT); Microsoft Corp, Res, Redmond//WA/98052; Univ Washington, Seattle//WA/98195; Princeton Univ, Princeton//NJ/08544 , 2003, V2578, P400-414

ISSN: 0302-9743 Publication date: 20030000

Publisher: SPRINGER-VERLAG BERLIN, HEIDELBERGER PLATZ 3, D-14197 BERLIN, GERMANYINFORMATION HIDING

Series: LECTURE NOTES IN COMPUTER SCIENCE

Language: English Document Type: ARTICLE

Geographic Location: USA

Journal Subject Category: COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE

Abstract: We describe a novel software verification primitive called

Oblivious Hashing. Unlike previous techniques that mainly verify the static shape of code, this primitive allows implicit computation of a hash value based on the actual execution (i.e., space-time history of computation) of the code. We also discuss its applications in local

software tamper resistance and remote code authentication.

Cited References:

*COMP CORP, SOFT DEB
AUCSMITH D, 1996, P 1 INT WORKSH INF
BARAK B, 2001, V2139, P1, SPRINGER VERLAG LECT
BLUM M, 1989, P86, P 21 ANN ACM S THEOR
CHANG H, 2001, P WORKSH SEC PRIV DI
COLLBERG C, 1998, P184, P ACM PRINC PROGR LA
COLLBERG C, 1998, P IEEE INT C COMP LA
COLLBERG C, WATERMARKING TAMPER
ERGUN F, 2000, V29, P1630, SIAM J COMPUT
ERGUN F, 1998, P259, P 30 ACM S THEOR COM
HORNE B, 2001, P WORKSH SEC PRIV DI
HUNT G, 1999, P135, P 3 USENIX WIND NT S
KNUTH D, 1973, V2, ART COMPUTER PROGRAM
MENEZES P, 1997, HDB APPL CRYPTOGRAPH
MONROSE F, 1999, P ISOC NETW DISTR SY
VENKATESAN R, 2001, P 4 INT WORKSH INF H
WANG C, 2000, CS200012 U VIRG
WASSERMAN H, 1997, V44, P826, J ACM

Author(s): Chen YQ (REPRINT) ; Venkatesan R ; Cary M; Pang RM; Sinha S ; Jakubowski MH

9/5,K/3 (Item 1 from file: 65)
DIALOG(R)File 65:Inside Conferences
(c) 2004 BLDSC all rts. reserv. All rts. reserv.

03965781 INSIDE CONFERENCE ITEM ID: CN041640182

A Graph Theoretic Approach to Software Watermarking

Venkatesan, R. ; Vazirani, V.; Sinha, S.

CONFERENCE: Information hiding-International workshop; 4th
LECTURE NOTES IN COMPUTER SCIENCE, 2001; VOL 2137 P: 157-168
Berlin, New York, Springer, 2001
ISSN: 0302-9743 ISBN: 3540427333
LANGUAGE: English DOCUMENT TYPE: Conference Papers
CONFERENCE EDITOR(S): Moskowitz, I. S.
CONFERENCE LOCATION: Pittsburg, PA 2001; Apr (200104) (200104)

BRITISH LIBRARY ITEM LOCATION: 5180.185000

NOTE:

Known as IH 2001

DESCRIPTORS: information hiding; IH

Venkatesan, R. ; Vazirani, V.; Sinha, S.